CSCI 435: Homework 3

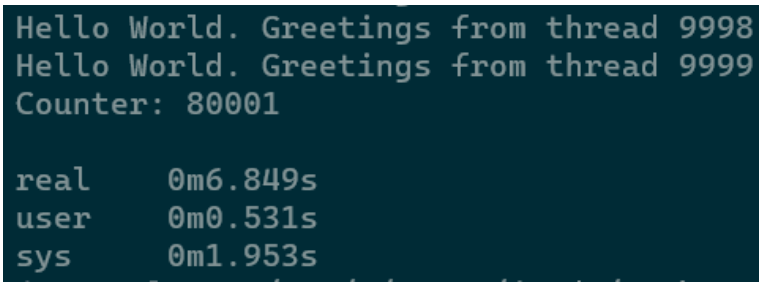
Jordan Taylor

March 1, 2022

For clarity, all experiments were run with 4 threads and 10,000 iterations. This would normally result with a counter value of 40,000.

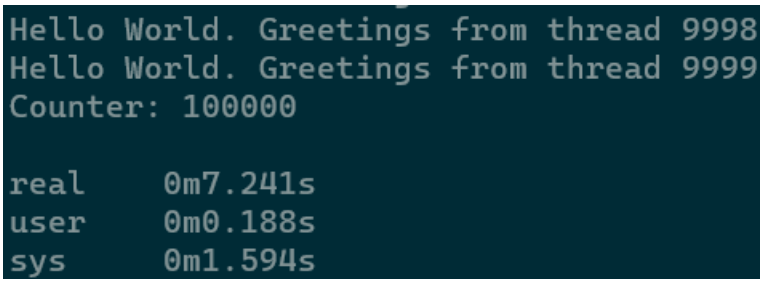
1. No synchronization

   In this experiment, the threads iterated at a seemingly random order. I thought it would have been faster since they didn't have to worry about each other but I am assuming it was slowed down because of race conditions, and the counter was unexpectly double my counter originally.



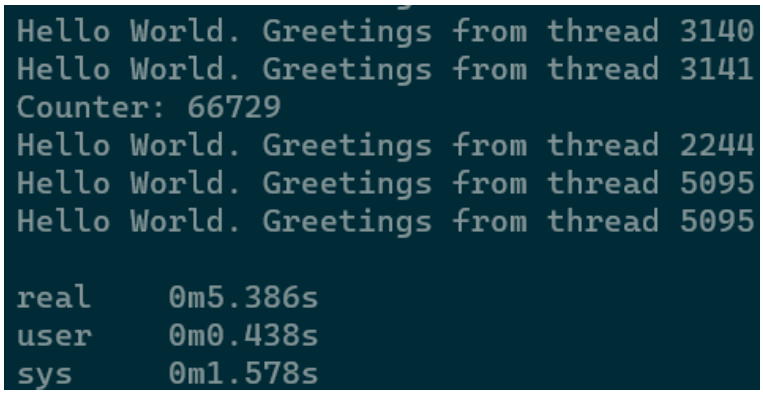
2. Mutex

   For this experiment, the counter was actually up to 100,000. This was unexpected and quite a bit more than the original counter value. The iteration seemed about the same as the unsynchronized test, but the time was a little higher.



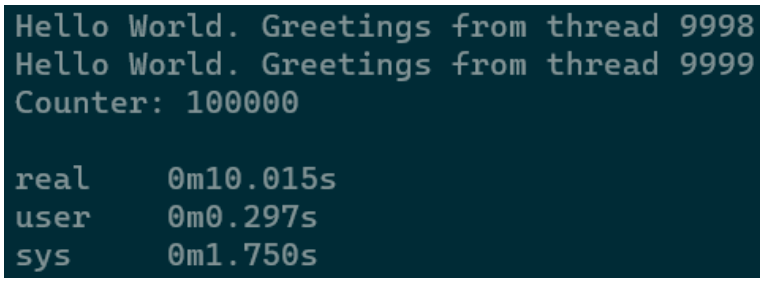
3. Test-And-Set

   The Test-And-Set experiment was similar, in coding terms, to the Test-Test-And-Set experiment (you'll see this one later). The counter value seemed random with this test, and I actually had to make the image a bit larger to capture the counter size. The time for it was a lot faster than the other 2 but I don't know if that really counts here if the counter was the lowest of them?



4. Test-Test-And-Set

   This experiment was clean and nice, aside from the time it took to execute. The iteration was synchronized and it paused in between each thread run. It was accurate with how it ran except for the counter value, which really threw me off. I can't seem to understand why exactly the counter value would be so high here.



5. Test with all experiments

   This was my favorite experiment because it was the fastest (well, beating test 1 by .001 second), and it provided an accurate count. However, the first test didn't have sleep included into the code. Therefore, actually this test was significantly faster but considering that sleep was require means that it's only a little bit faster (it's all really confusing).

```
Hello World. Greetings from thread 9998
Hello World. Greetings from thread 9999
Counter: 40000

real    0m4.010s
user    0m0.141s
sys     0m0.750s
```